1 at step 102, corresponding to each of the respective items or groups of data. Information objects for each of the ribbons in ribbon group 1 are then created at 104, and information objects are attached to each of the ribbons in ribbon group 1 at 106. A similar process is followed in relation to ribbon groups 2 and 3.

When all of the intersection details for the ribbon groups 1 and 2 have been obtained at 400, the intersection objects for ribbon groups 1 and 2 are created at 402 together with an indication of the intersection type. An information object for each intersection is created at 404 and the information objects are attached to the intersection objects at 406. A similar process is adopted once the intersection details for the ribbon groups 1 and 3 have been obtained at 500. Once the information objects have been attached to the ribbons in groups 1 and 2 and to the intersection objects, the DataWeaver software creates a weave object at 600, and attaches the group 1 ribbons to the weave horizontally at 602. The group 2 ribbons are attached to the weave vertically at 604 and the ribbon group 1 and 2 intersection objects are attached to the weave at 606. The completed weave of ribbon groups 1 and 2 is then attached to the map at 800 and its position relative to other weaves on the map indicated. A similar process is adopted to create a weave object at 700, once the information objects have been attached to the ribbons in ribbon groups 1 and 3 and to the intersection objects. In this particular weave map, the group 1 ribbons always appear horizontally, whereas group 2 and group 3 ribbons appear vertically in the completed map. The weaves are arranged to determine their absolute position on the map at 802 and the map is then ready for display on a display screen at 804.

In addition to serving as a static graphic means of describing a topography of relationships and particular intersections, the DataWeaver method also lends itself to use as a graphic user interface (GUI) for computer programs dealing with information. The nature of the graphic image coincides with the manner in which program architectures are constructed, thus assisting in their preparation and increasing the ease of integration of superstructure and

substructure. The particular program architecture employed will depend to some extend on the specific application in which the DataWeaver method is embodied.

In order to further illustrate the nature of the invention a preferred embodiment of the DataWeaver method in the medical field will now be described with reference to Figures 4 to 6 in which a process by which the clinical data and encounters of a patient with the professional health care system are managed. In this embodiment, the DataWeaver software is embodied in a graphical user interface which is used in conjunction with other known software in clinical knowledge databases, patient records, etc. The DataWeaver method is not embodied in the software that underlies the actual graphical user interface (GUI) construction, which follows standard professional software practice. Figure 5 illustrates how a typical client/server clinical system which employs DataWeaver as a front end at the GUI may be set up. The DataWeaver front end communicates with the local clinical objects or the remote clinical objects to store and retrieve information in the databases 34 via servers 32.

Figure 4 illustrates an ideal software architecture for a clinical system employing DataWeaver which typically consists of five layers with client-server relationship between layers. A layered approach was chosen to build the system in order to make construction of the weave map modular and de-coupled from the underlying clinical system. The first layer at the front end is the GUI 40 which embodies the DataWeaver method. The second layer is the GUI/domain control layer 42 which assembles the map, communicates with the supporting clinical domain control layer 44 and provides the map with data and control logic. This second layer 42 defines the semantics and look and feel of the map (e.g., what does an intersection mean, what icons are used, definition of areas in map, etc.). The domain and services layer 44 consists of various domain objects and services implemented as components that can be distributed across several machines (e.g., patient records, treatment planning engine, log keeping and access control, etc.). The persistence management layer 46 takes care of storage and retrieval

of information in a platform and vendor independent way. The final layer 48 consists of the back end databases. The above-described layered approach decouples the DataWeaver method from the underlying clinical system and ensures the reuseability of the DataWeaver widgets in other applications within the medical field as well as in various other complex business, planning, engineering or scientific environments outside of the medical field.

The various events that take place when a patient visits a hospital for medical help constitute a clinical encounter. The patient is taken through a series of processes and is treated and monitored for various conditions that he/she may have. Fundamentally there is a work flow (a series of processes) and the data involved in this exercise. The most common steps are:

(a) register the patient
(b) interview the patient or responsible party
(c) acquire data on various signs and symptoms
(d) do a diagnosis based on what is acquired
(e) treat the patient based on the diagnosis/condition
(f) monitor various conditions/parameters

Figures 6 to 10 illustrates how the GUI employs the described embodiment of the DataWeaver method to present this series of processes and the associated data and relationships on a single page map for display on a video display unit (VDU). A hard copy of the display can also be printed in full colour on paper if required. The map is constructed dynamically by interacting with the user and the clinical objects. The overall topology and direction of process flow, however, is decided early in the program. The map shown in Figure 6 is a fully-grown map which grew in order from A to E as described below. The map consists of areas (portions boxed in by dotted lines, which need not be part of the active display) and weaves. An area comprises weaves related to the same process. A weave is a collection of two groups of ribbons, their intersections, and associated "information" and "action" objects. Information